



DIME DYNAMIC DOCUMENTATION TRAINING

Exercise 2

Luiza Andrade & Mrijan Rimal

June 8, 2017

Introduction

Exercise 1 introduced you to the basics of how to import tables and figures to a \LaTeX document. This exercise will introduce you to some intermediate topics commonly used to make your document look even more professional.

We will also show how \LaTeX can be used to create a dynamic document that updates automatically once your output from, for example, Stata or R is updated without any error prone manual copy-and-pasting.

Part 1. Exporting tables

In this section, you have the option of using the template do file provided in [Dynamic-Documentation/Exercises/Stata Export Exercise/Export tables and images.do](#) to export graphs and figures OR using your own data and generating tables and graphs that can be imported into \LaTeX .

If you're using the do files provided, change the folder paths in the do file to your directory structure and run the do-file. This will export two tables and two graphs to the Raw folder that will be used in the exercises that follow.

If you would like to work with your own data, please follow the following steps.

- Make sure that you export **two tables** and **two graphs** using your own data. For help, you can look at the folder called “Exercise Stata - How to export tables and graph from Stata to LaTeX”.
- Make sure that all the exporting is done using Stata do-files, or R scripts so you can easily export them again by just running the code.
- Import all the tables and figures you have created in step 1 into your \LaTeX document. Sample code to import tables and figures can be found in the **DIME Templates** sub-folder. Make sure to use the code in **Template 1 - Importing tables.tex** file to import figures(not tables), and the code in **Template 2 - Importing figures.tex** file template to import graphs(not tables). It will not work if it is the other way around.
- Make sure you give a caption to your tables.

Part 2. Intermediate L^AT_EX Exercises

Part 2.1 Adding Sections to your document

L^AT_EX automatically formats the document and different section headers and subheaders according to predefined formats. It also automatically creates beautiful **Table of Contents** based on what has been defined as sections and subsections.

Sections can be created using `\section{title}` command. L^AT_EX automatically numbers all the sections in the order you put them. Since, you manually don't specify chapter numbers, you can cut and paste subsections from the end to the beginning and the numbering will update automatically!

Similarly, subsections can be created using `\subsection{title}` and sub-sub-sections can be created using `\subsubsection{title}`. The subsections will be numbered 1.1 and the subsubsection will be number 1.1.1 if created within the first section. An example is shown in Annex 1.

Note: Sections and subsections can be created using `\section{title}` if you do not want your sections numbered in the document. However, these sections and subsections will not be shown in the table of contents.*

Part 2.2. Adding Table of Contents

After we have set up sections, sub-sections and sub-sub-sections you can easily add a table of contents to your L^AT_EX document by using `\tableofcontents` in your document. You can create this anywhere you want, but typically this is created directly after `\begin{document}` or after `\maketitle` if you have a title. An example is shown below:

```
\documentclass[12pts]{article}
```

```
\tableofcontents
```

```
\newpage
```

```
%opening
```

```
\title{My Awesome Document}
```

```
\author{John Doe}
```

The above L^AT_EX code is used to generate table of contents.

```
\begin{document}
```

```
\maketitle
```

My Awesome Document

John Doe

May 31, 2017

Contents

1	This is a test section	2
2	This is the second test section	2
2.1	This is a test subsection	2
2.1.1	This is a test subsubsection	2

Figure 1: Example table of contents generated from our document in Annex 1.

Part 2.3. Referring to your tables, and figures in a document

L^AT_EX uses a dynamic referencing system for all our tables, figures, sections, subsections, etc. This means that they will be numbered in whatever order you put them in. For example - if you cut(as in cut paste) and paste a table that was the ninth table in the document and put it first, the table number will automatically update to 1 for that table. So, rather than telling the user to look at a table by referring it with a number, you can give your tables and graphs a name, and reference it using the commands you'll learn in part 2.3.1 and 2.3.2, so if you move the tables around, the numbers that they will see in the pdf document will still correspond to the actual number of the table.

In this subsection, we will learn how to reference to a table, or a figure. Figures and tables have different referencing system that we will learn in part 2.3.1 and 2.3.2 respectively.

Part 2.3.1 Referring to Figures in a document

To refer to figures, we will use `\label{fig:figurename}` command inside our begin and end figure like the example below. To add a label to the imported `iegraph` figure, we just add the lines `\label{fig:Figurename}` inside the `begin` and `end` figure in our codes.

```
\begin{figure}[H]
\includegraphics[width=\textwidth]{../Raw/iegraph.png}
\caption{Add figure title here}
\label{fig:iegraph}
```

```
\end{figure}
```

Now to refer to the figure anywhere in the document, you can type `\ref{fig:iegraph}` and it will automatically refer to the figure using the correct number in the document. In our example, if the `iegraph` figure is the fourth one in the document, `As you can see in figure \ref{fig:iegraph}` will appear as `As you can see in figure 4`, in the document.

Part 2.3.2 Referring to Tables in a document

Similarly for tables, we can also refer to them anywhere in the document using the same method. To refer to tables, we will use `\label{tab:tablename}` command inside our `\begin{table}` and `\end{figure}` like the example below.

First, we need to add a label to the table that has been imported to the document by adding `\label{tab:Tablename}` inside the `begin` and `end` table in our code.

```
\begin{table}[H]
\caption{Add a title to this table}
\input{../Raw/sample_sizes.tex}
\label{tab:samplesize}
\end{table}
```

Now we can refer to the above table in our document by typing `\ref{tab:samplesize}` which will automatically refer to the table using the correct number in the document.

Part 2.4. Line Spacing

Now that you have created sections in your document, and added text, you might want to change the spacing between lines. This can be achieved using the `setspace` package.

The steps to change line spacing are as follows:

- Before `\begin{document}`, declare the package you will be using. In this case, `\usepackage{setspace}`. This package controls the line spacing in a \LaTeX document.
- After using the `setspace` package, you should now specify line spacing which are as follows:

singlespacing This option specifies the document to use one line spacing.

doublespacing This option specifies the document to use two line spacing.

onehalfspacing This option specifies the document to use one and a half line spacing.

```
\usepackage{setspace}
\doublespacing
%\onehalfspacing
%\singlespacing
\begin{document}
\maketitle
\listoftables
\listoffigures
\section{Introduction}
This is the introduction paragraph
```

The **setspace** package can also be used to specify more particular linespacing i.e. 1.7 lines, 2.4 lines etc using the **setstretch** feature but that will be covered in another documents.

Part 2.5. Rotating a table landscape

Sometimes the tables are very wide and need to be in landscape format. This can be adjusted using the **adjustbox** package which we have been using for importing tables.

```
\begin{table}[H]
    \caption{Add table title}
    \input{sample_sizes.tex}
    \label{tab:samplesize}
\end{table}
```

To rotate the table to landscape, we can use the **adjustbox** feature with the **angle = 90** option as shown in blue in the code below.

```
\begin{table}[H]
    \caption{Add table title}
```

```
\begin{adjustbox}{angle = 90}
    \input{sample_sizes.tex}
\end{adjustbox}

\label{tab:samplesize}

\end{table}
```

This will rotate the table to the specified degree in the final document.

Part 3. Making a Dynamic Document

Here, we will produce a dynamic document. Please only do this if you have completed up to Part 2.5 of this exercise document.

1. Generate a pdf document by compiling the document and save the pdf created up to now in a separate location from the `.tex` document where you have input the codes to import the tables and graphs.
2. Generate a new treatment for your dataset by dropping a subset of your data set.

If you are using the Stata do-file template provided in the Dynamic Documentation Folder, then only keep observations from Europe and Central Asia, and drop observations from North America and South America. This can be done by using the Stata codes `drop if region!= 1`.

3. Rerun the do-file . Make sure that the tables and graphs you created earlier are overwritten i.e. save them without updating any paths so they save and replace the tables and graphs created earlier.
4. Now, open the L^AT_EX file¹ you created earlier and press *Build and Compile* under the *Tools*.
5. Now if you compare the pdf file you have just generated with you new data with the one you saved earlier, you will find that the tables would have updated automatically.

¹The one where you imported all your graphs and figures.

Part 4. Challenge: Using a do-file to edit a .tex file after exporting it

Note: This is an advanced exercise and you don't have to feel compelled to finish it. This is also best done after finishing all the other exercises in the GitHub repository.

During this part of the exercise, you will learn how to use commands in Stata to format your tables. The reason for doing this is that we want to make as little edits in L^AT_EX as possible, so that when we update the tables like we did in Part 3, everything in the document file updates automatically. So, we would want to tweak or format our tables automatically using Stata, so that we do as little work in L^AT_EX as possible.

While tables exported from Stata to L^AT_EX are generally very nice, sometimes they need to be tweaked a little to make them look nicer. So, in this exercise, we'll use the `filefilter` command in Stata to make small changes to the files exported by Stata. `filefilter` is one such command in Stata which used to edit output files generated by Stata.

This exercise requires more familiarity with L^AT_EX than previous exercises. Don't worry if you can't complete it.

Task 1: Run the initial code for exercise 6 in the [Add path and name of do-file](#) do-file. This will create a table with sample sizes for control and treatment groups across regions and in the whole sample. Add this table to the .tex file you created in the Exercise 2, Part 1. How does that look?

Task 2: Open the [Dynamic-Documentation/Exercises/Exercise 1/Output/Raw/sample_sizes.tex](#) file created by Stata. Can you identify the source of the extra spacing?

Task 3: Use the `filefilter` command in Stata to filter out the lines or characters in the fragmented file that create the extra spacing. Import the new .tex file and check how it looks.

Task 4: Repeat task 3 if necessary.

Annex 1 : Creating sections

Commands typed in LaTeX document.

```
\section{This is a test section}
Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque
laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi
architecto beatae vitae dicta sunt explicabo.

\section{This is the second test section}

\subsection{This is a test subsection}
Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci
velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam
aliquam quaerat voluptatem.

\subsubsection{This is a test subsubsection}
Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit
laboriosam, nisi ut aliquid ex ea commodi consequatur?
\end{document}
```

Output

1 This is a test section

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo.

2 This is the second test section

2.1 This is a test subsection

Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem.

2.1.1 This is a test subsubsection

Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur?

Figure 2: Adding sections and subsections in a L^AT_EX document